Dr. Michael Eichberg
Software Technology Group
Department of Computer Science
Technische Universität Darmstadt

Introduction to Software Engineering

Organization

TECHNISCHE UNIVERSITÄT DARMSTADT

Teaser

TECHNISCHE
UNIVERSITÄT
DARMSTADT

> *As long as there were no machines, programming was no problem at all; when we had a few weak computers, programming became a mild problem, and now we have gigantic computers, programming has become a gigantic problem.*

E.W. Dijkstra

The Humble Programmer
Communications of the ACM, Vol. 15, Issue 10, 1972

The Team

TECHNISCHE UNIVERSITÄT DARMSTADT

The Team

Dr. Michael Eichberg

Sven Amann

Manuel Weiel

Contact

WWW

http://www.stg.tu-darmstadt.de/teaching/teaching_iverview.en.jsp

Forum

http://www.d120.de/forum/

→kanonische Einführungsveranstaltungen

→Einführung in Software Engineering

Organization

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Lecture

- Thursdays 13:30-15:00 in S2 02 | C205 and C110
- The slides are in English
  (Key terms will be translated into German.)
- The slides will generally be available after the lecture
  (I will try hard to make a preliminary version available the day before the lecture.)
- The slides can be found at:
  http://stg-tud.github.io/eise/

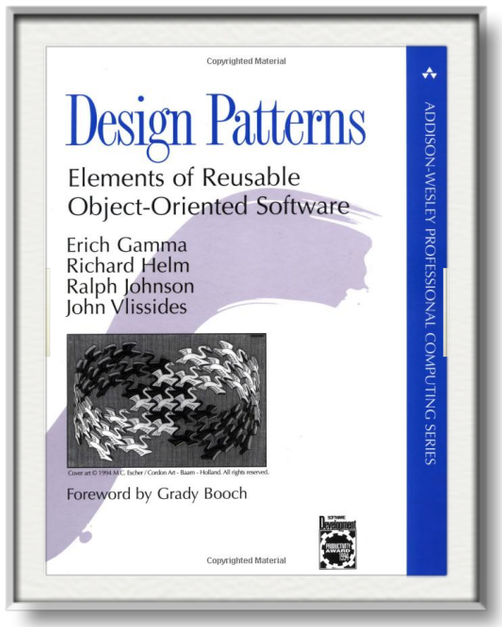
Exercises

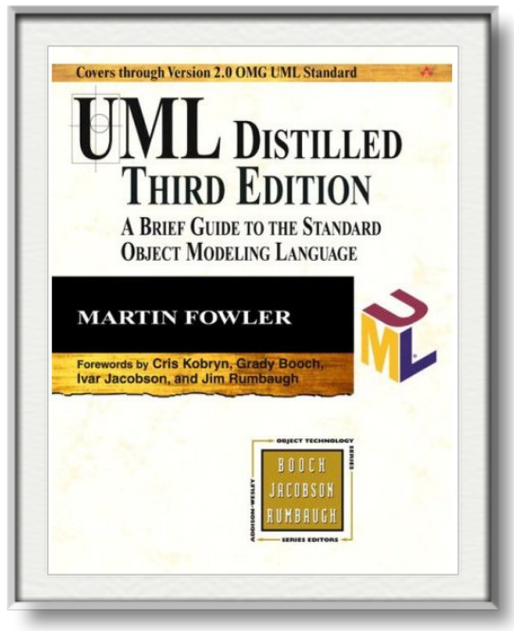
- Every week, we will have an exercise, starting **today**
- Exercises are expected to be solved in teams of 3 students
- The content of the exercise is relevant for the exam
- Sign-up as a team; if you don't have a team, **we will** assign you to a team
- All further details will be announced in today's exercise.
- **The content of the exercise is relevant for the exam**

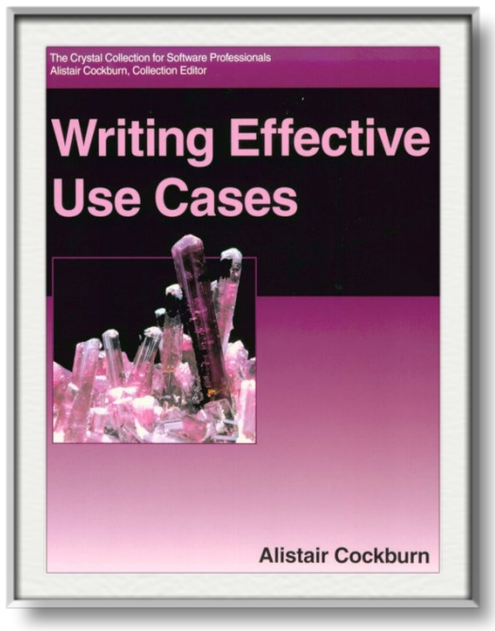
Written Exam

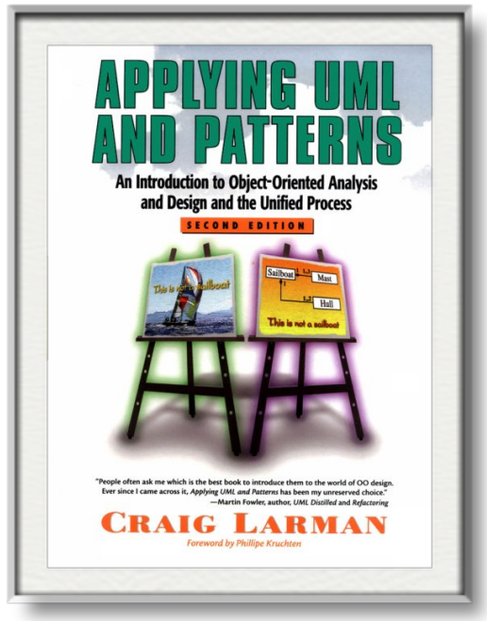
- The exam will be on February, 26th 2015 - 14:30 - 16:00
  (The room will be announced in due time. The exam will take 90min.)
- You need to register for the exam in TUCaN
  (There are no further prerequisites; “everyone” can attend the exam.)
- The exam will be an open book exam
- The exam will enable you to choose the tasks that you are particular good at
  (I.e., it will not be possible to solve the entire exam in 90 minutes.)

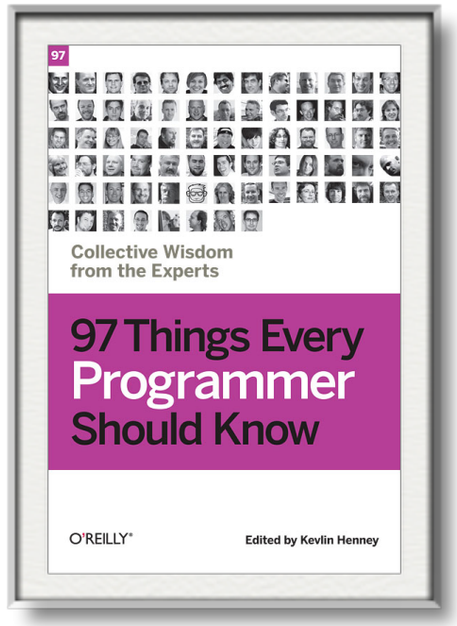
Related Bibliography

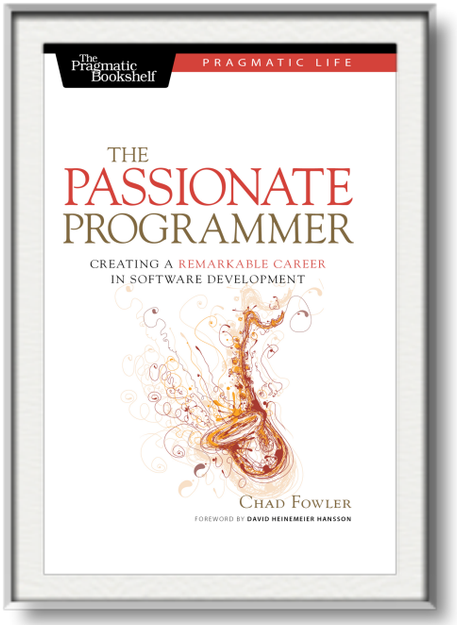








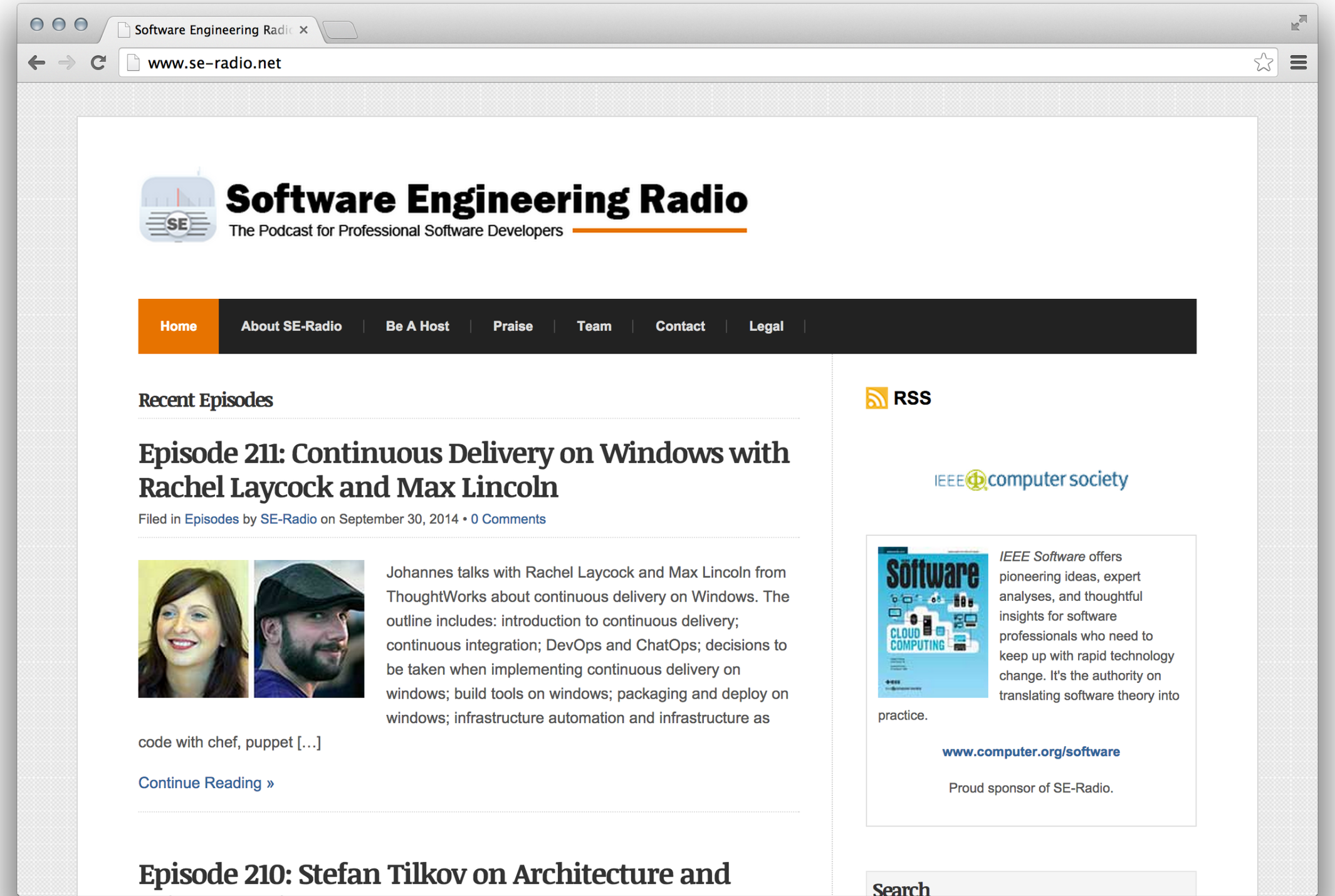




Essential Bibliography

- Design Patterns - Elements of Reusable Object-Oriented Software; Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides; Addison-Wesley, 1995
- Applying UML and Patterns - An Introduction to Object-oriented Analysis and Design; Craig Larman; Prentice Hall

A Recommended / Very Useful Podcast



External Talks / Events

- **MSG**
  “Requirements Engineering”
  November, 2014
- **Capgemini**
  Excursion / One-day Workshop
  December, 2014
- **ACCSO** - Working Title:
  “Beschleunigte Softwareentwicklung”
  January, 2015

The Lecture

TECHNISCHE
UNIVERSITÄT
DARMSTADT

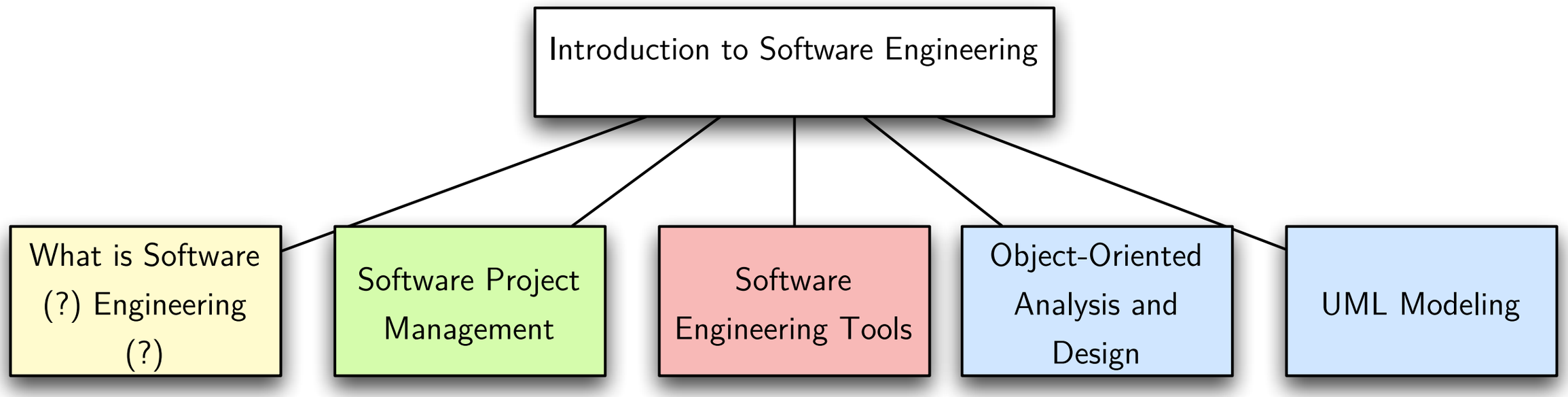
Basic programming skills are required.

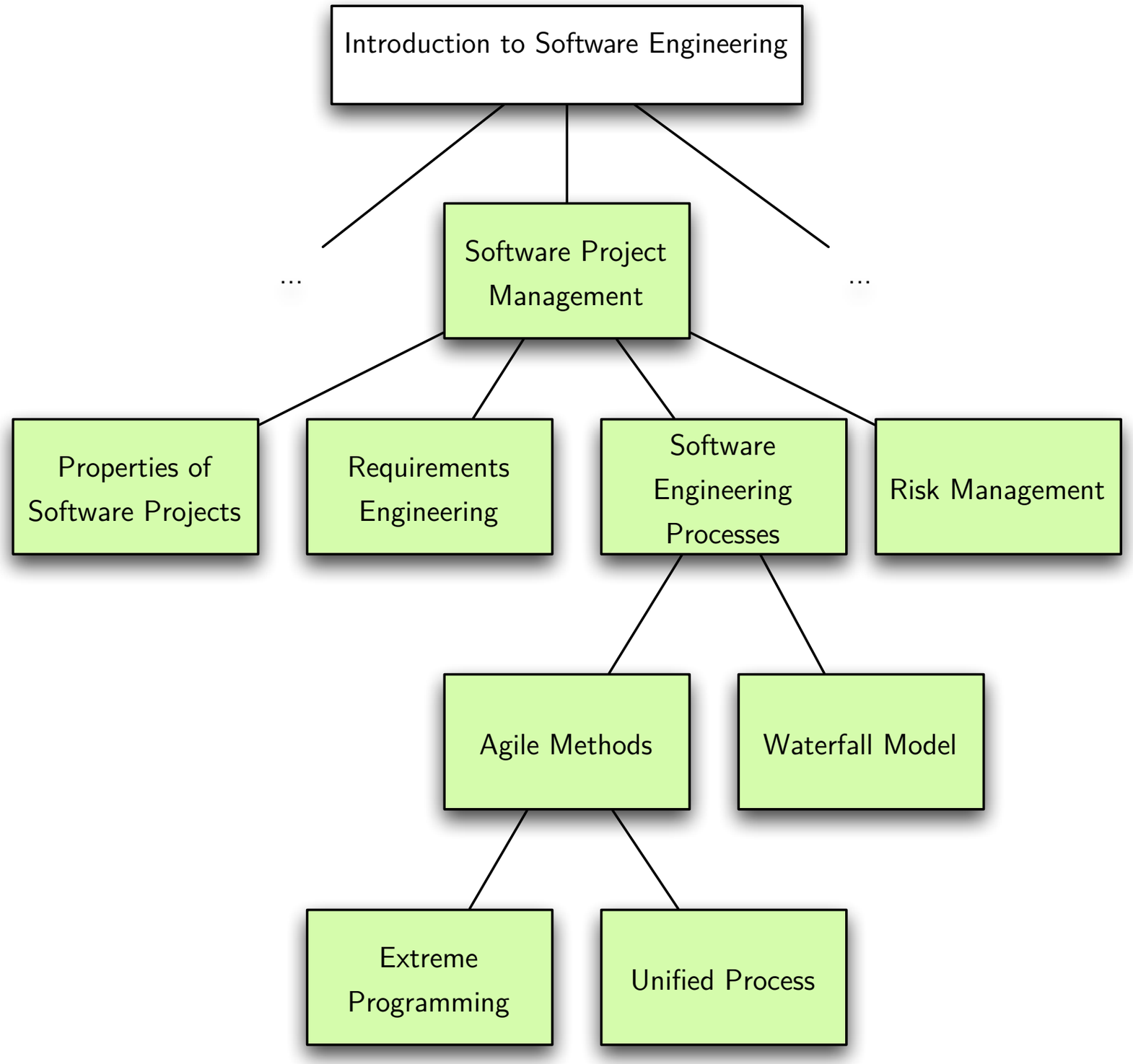
- Basic knowledge of **object-oriented programming** concepts is necessary
  I.e., you should readily understand the following terms:
  - class, interface
  - object
  - inheritance
  - polymorphism
- Working knowledge of the **Java** (7) programming language
- We are going to use Java 8

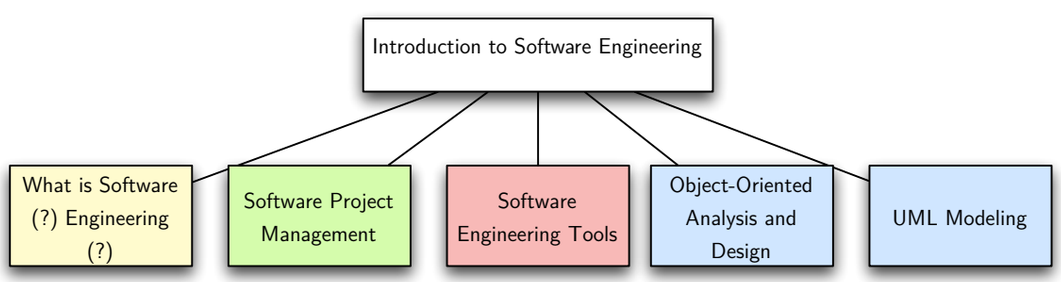
Basic Goals

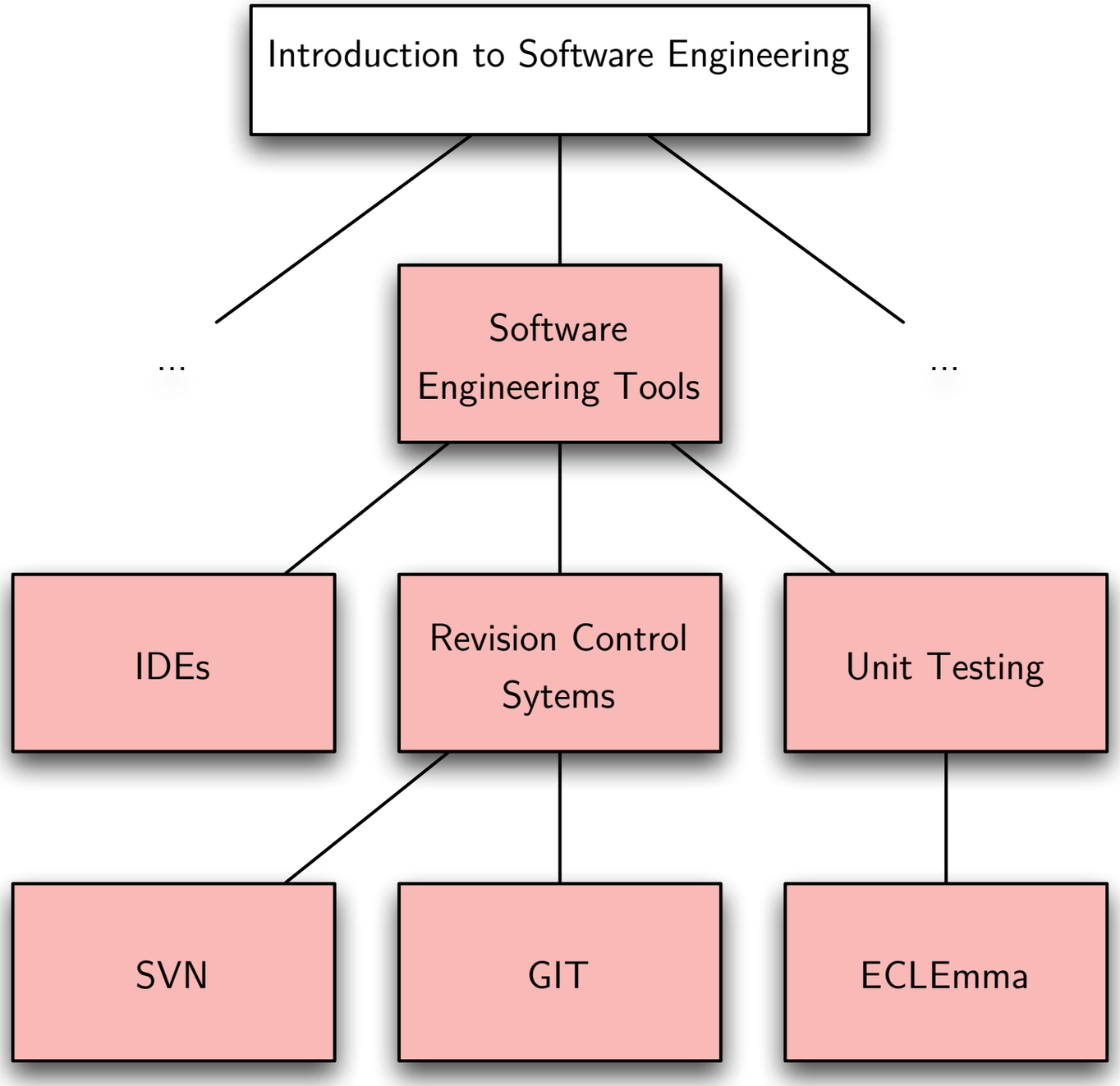
- To get a brief overview of “all” areas of software engineering
- To understand agile software development processes
- To be able to perform object-oriented analysis and design
- To be able to read and create basic UML diagrams
- To perform basic Software Quality Assurance
- To get first hands-on experience and to learn to use basic software development tools

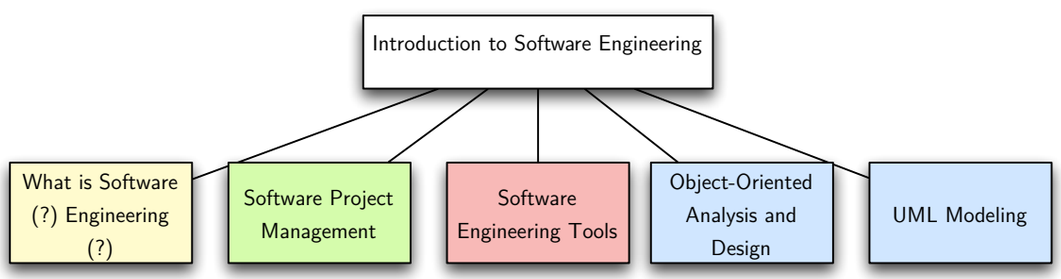
The goal is to enable you to systematically carry out small(er) commercial or open-source software projects.

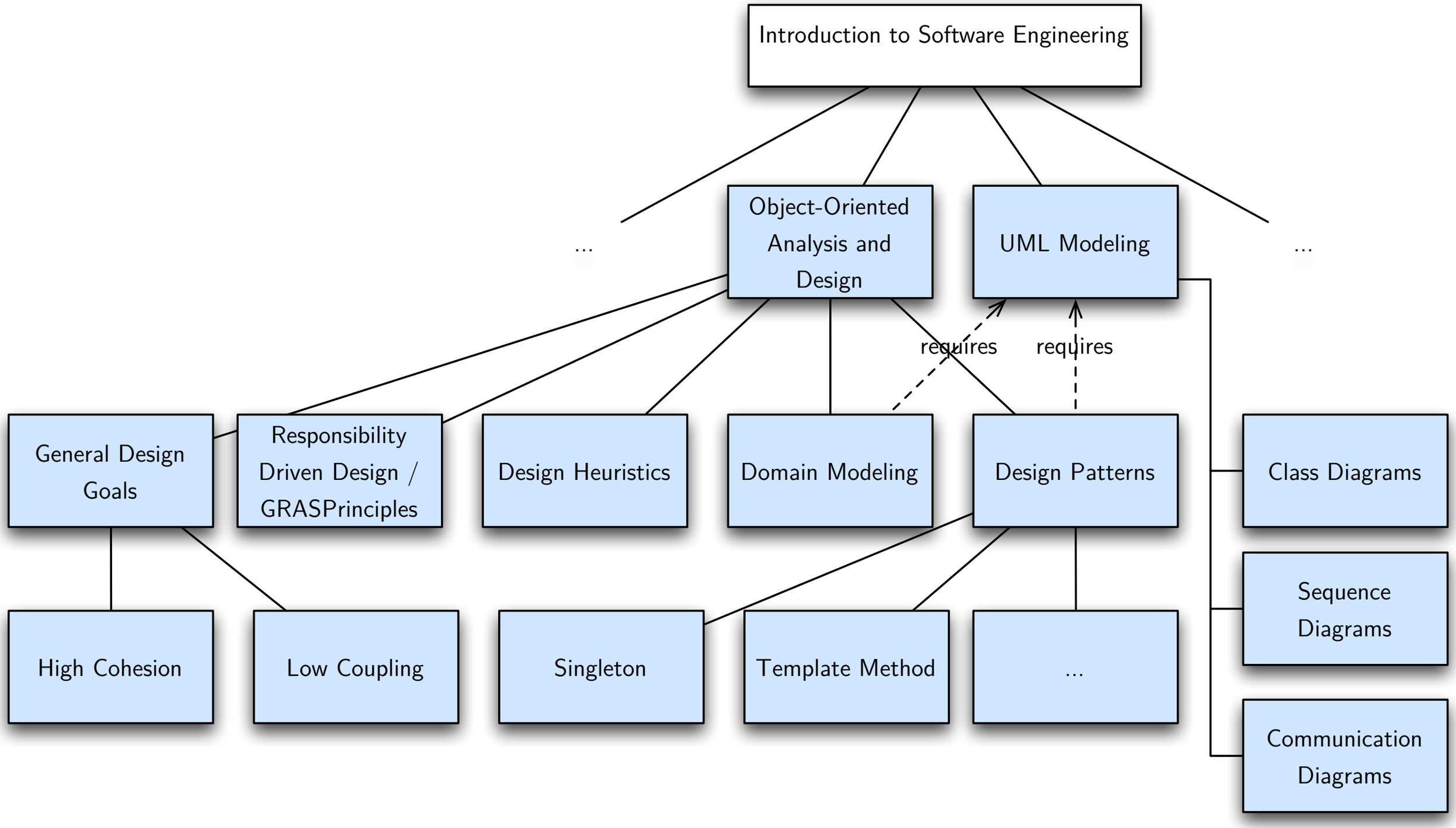

Introduction to Software Engineering
What is Software (?) Engineering (?)
Software Project Management
Software Engineering Tools
Object-Oriented Analysis and Design
UML Modeling

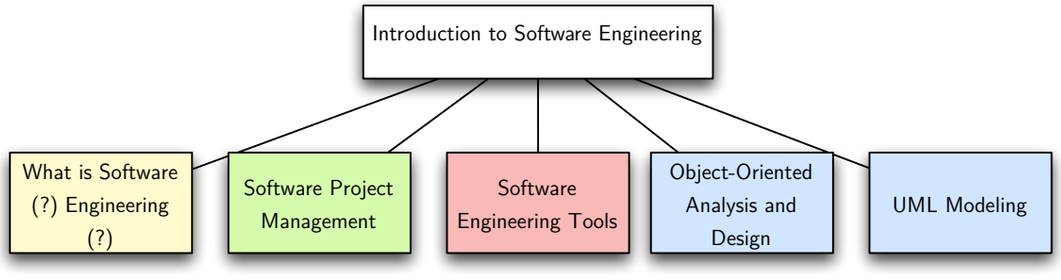

Introduction to Software Engineering
...
Software Project Management
...
Properties of Software Projects
Requirements Engineering
Software Engineering Processes
Risk Management
Agile Methods
Waterfall Model
Extreme Programming
Unified Process

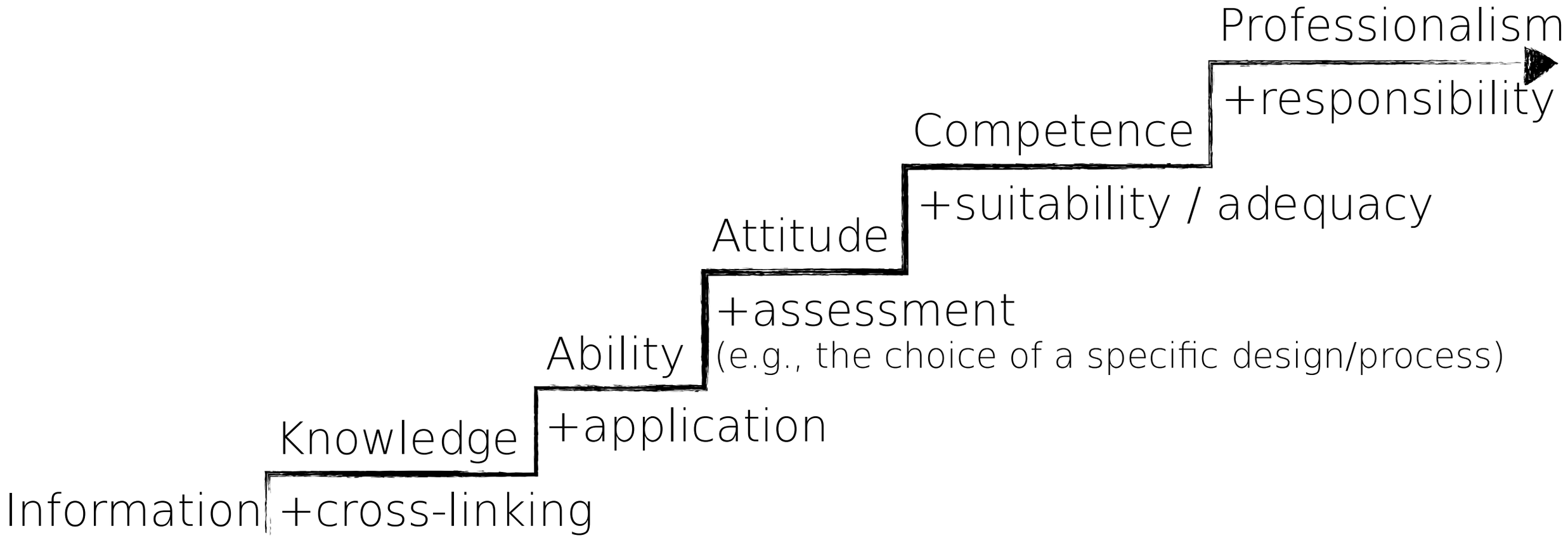

Introduction to Software Engineering
What is Software (?) Engineering (?)
Software Project Management
Software Engineering Tools
Object-Oriented Analysis and Design
UML Modeling


Introduction to Software Engineering
...
Software Engineering Tools
...
IDEs
Revision Control Sytems
Unit Testing
SVN
GIT
ECLEmma

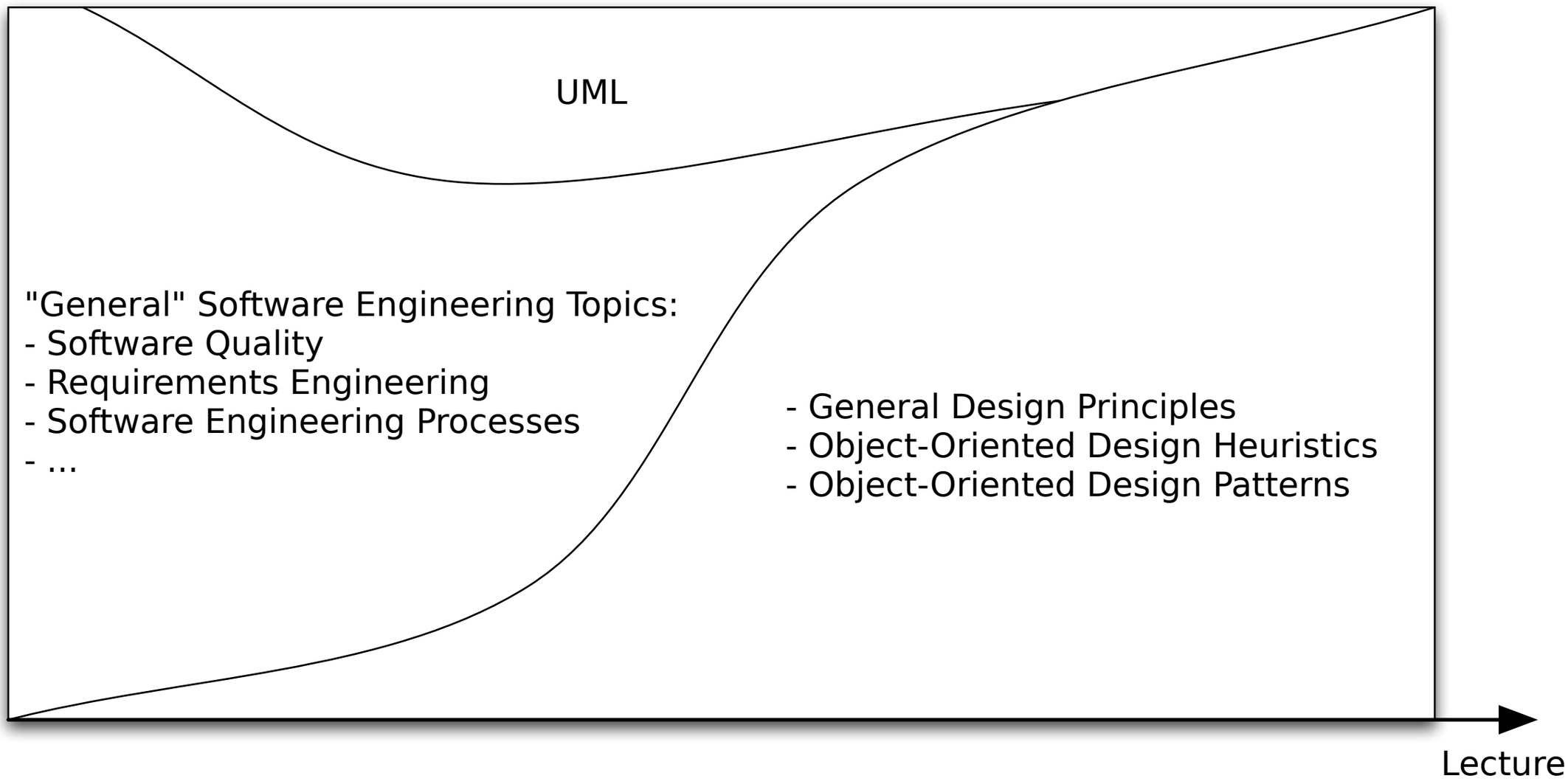

Introduction to Software Engineering
What is Software (?) Engineering (?)
Software Project Management
Software Engineering Tools
Object-Oriented Analysis and Design
UML Modeling

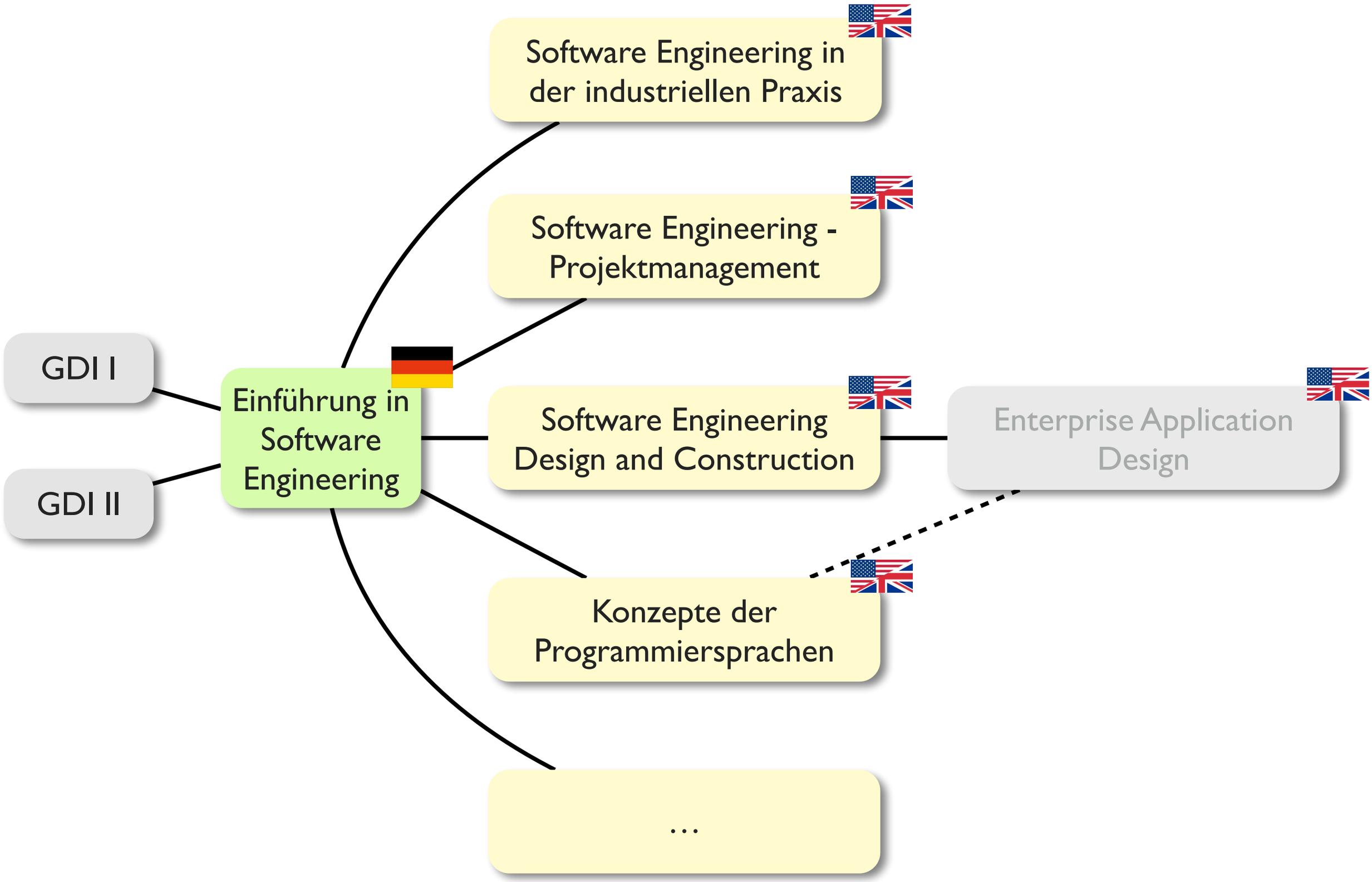

Introduction to Software Engineering
...
Object-Oriented Analysis and Design
UML Modeling
...
requires
requires
General Design Goals
Responsibility Driven Design / GRASPrinciples
Design Heuristics
Domain Modeling
Design Patterns
Class Diagrams
Sequence Diagrams
Communication Diagrams
High Cohesion
Low Coupling
Singleton
Template Method
...


Introduction to Software Engineering
What is Software (?) Engineering (?)
Software Project Management
Software Engineering Tools
Object-Oriented Analysis and Design
UML Modeling


Information
+cross-linking
Knowledge
+application
Ability
(e.g., the choice of a specific design/process)
+assessment
Attitude
+suitability / adequacy
Competence
+responsibility
Professionalism

Content Distribution




Software Engineering in der industriellen Praxis
Software Engineering - Projektmanagement
GDI I
Einführung in Software Engineering
Software Engineering Design and Construction
Enterprise Application Design
GDI II
Konzepte der Programmiersprachen
…